SE9-99-012

1963-7341

5   A SYSTEM AND METHOD OF ENRICHING NON-LINKABLE MEDIA
REPRESENTATIONS IN A NETWORK BY ENABLING AN OVERLYING HOTLINK
CANVAS

Inventors: Sih-Pin S. Chang, Shu-Chun Jeane Chen, K. Kumar, j. Lipscomb, J. Menon, and L-J
10   Zhang

Related Applications

1. Serial No. _____ entitled "Framework for Progressive Hierarchical and Adaptive Delivery
15   Rich Media Presentation and Associated Meta Data", filed _____ (SE9-98-033).

2. Serial No. _____ entitled "Personalizing Rich Media Presentations Based on User Response
to the Presentation", filed _____ (SE9-98-028).

20   3. Serial No. 09/100,418 entitled "Progressive to Relieve Delivery of Interactive Descriptions
and Render it for Electronic Publishing of Merchandise" filed June 20, 1998 (SE9-98-004).

4. Serial No. _____ entitled "System and Method for Tracking User Interations and
Navigation During Rich Media Presentations", filed _____ (SE9-99-011/1963-7344).

25

5. Serial No. _____ entitled "Method for Non-linear Transformation and Interpolation for
Unifying Hotlinks Between Heterogeneous Media Representations", filed _____ (SE9-99-
004).

30   All of the above-related applications are assigned to the same assignee as that of the
present invention and are fully incorporated herein by reference.

1

9954_1

BACKGROUND OF THE INVENTION

(1) FIELD OF THE INVENTION

5

This invention relates to multimedia networks, systems and methods of operation. More particularly, the invention relates to systems and methods for enriching non-linkable media representations in a network by enabling an overlying hotlink canvas.

10   (2) DESCRIPTION OF PRIOR ART

In recent years, there has been a sizeable growth in the use of rich media over the World Wide Web as more and more individuals and institutions are beginning to realize the web's potential in a broad range of applications including electronic commerce, education training, news, etc. Examples of rich media include animation, audio, 3-D, panoramas, and videos. There are two apparent clusters of rich media technology. One at the low end and the other at the high end which hampers effective deployment of some rich media in Internet based applications. The "low end" cluster comprises static images and simple non-interactive animations (typically animated GIFs) which are easy to deploy and therefore have widespread usage. The "high end" clusters comprises richer and more natural experiences with larger interactivities, such as panoramas, 3-D, streaming audio/video, and composite media (e.g., MPEG-4), but the difficulty of deployment has limited widespread usage. A novel scaleable architecture called HotMedia bridges the gap between the two clusters thereby achieving widespread web penetration. A key feature of the technology is a suitable delivery file format that can contain heterogeneous compositions of media bit streams as well as meta data that defines behavior, composition and interaction semantics. The delivery file format enables the creation of lightweight single file representation of interactive, multistage presentations resulting in multiple media type contents. At the core of HotMedia client is a smart content algorithm that infers media types from the incoming data stream and fetches the media renderer components, user-interface components and hyper-linked action components, all just in time, resulting in progressive and context driven enrichment of the user experience. Further details related to HotMedia architecture are described

9954_1

in Serial No. _____ entitled "Framework for Progressive Hierarchical and Adaptive Delivery Rich Media Presentation and Associated Meta Data", filed _____ (SE9-98-033), supra and in IBM Research Report 21519[97069] entitled "The Hot Media Architecture: Progressive & Interactive Rich Mediafor the Internet" by K. G. Kumar et al, dated July 1999.

Often instances of rich media incorporate links to other presentations to expand a user experience. The process of clicking on a link in a media for transfer to the other presentation is referred to as "hotlinking "or "hyperlinking" which is further described in USP 5,841,978 entitled "Networking Using Steganographically Embedded Data Objects" issued November 24, 1998 and USP 5,918,012 entitled "Hyper Linking Time Based Data" issued January 29, 1999. Sometimes rich media are non-linkable to other presentations. It would be desirable to have non-linkable rich media supplemented to provide hotlinking to other presentations. Alternatively, it would be desirable to de-couple hotlinking from media.

Typically, hotlinks have been tightly integrated into supported media. For example, NetShow available from Microsoft supports hotlinks in the temporal domain. All hotlinks are written to its AFS file format along with the media. RealNetwork G2 uses Synchronized Multimedia Integration Language (SMIL)(http://www.trainingsupersite.com/realplayerG2/) for a composition of temporal hotlinks. Neither RealNetwork nor NetShow support hotlinks other than in the temporal domain. Veon's (http://www.veon.com/v-active) with integration of RealNetwork G2 provides both spatial and temporal hotlinks uses SMIL as output format. None of the above-mentioned supported media provide a generic plug and play framework for non-linkable media to become hyperlinkable. Nor does such hotlink supported media provide, in real-time, a separate hotlink meta data comprised in advance and delivered in addition to a transparent panel as a hotlink canvas for implementation of hotlinking in otherwise non-linkable media.

SUMMARY OF THE INVENTION

An object of the invention is a multimedia network and method of operation for enriching non-linkable media representation for hyperlinking.

3

9954_1

4

Another object of the invention is a multimedia network and method of operation providing a generic plug and play framework for non-linkable media to become hyperlinkable.

5   Another object is a multimedia network and method of operation providing a hotlink canvas as an overlay on non-linkable media for hyperlinking.

An object is a multimedia network and method of operation providing an overlay canvas for decoupling hotlinks in accompanying media.

10

An object is a hotlink canvas as an overlay in a multimedia network composed in advance and delivered with rich media for implementing hotlinking in accompanying non-linkable media or de-coupling hotlinking in accompanying media otherwise linkable.

15   These and other objects, features and advantages are achieved in a network capable of providing rich media presentations. The network includes a network server and a client station and implements HotMedia architecture to transmit a presentation in rich media including non-linkable media from the server to the client station. The server transmits the presentation as a streaming file in a framework of frames including a header frame, thumbnail frame, meta frame, media frame and end-of-stream frame. The client station includes an action enabling kernel and

20   a client master module. The action enabling kernel contains a table lookup of action triggering inputs and consequent action commands. The master module controls a selection process of the various types of media involved in the presentation. When a media frame is encountered the master module checks to determine that a media object has been created to handle the media

25   frame. If not, the master fetches the media object from the server and creates an instance of the media object to render the media frames. The rich media file may also include meta frames with range and action subtypes for the media. The range and action frames are the meta data for hotlink compositions and actions. When the first meta frame is received, the master fetches the action enabling kernel and a hotlink canvas from the server. The master delivers the meta frames

30   to the action enabling kernel. The hotlink canvas receives hotlink composition from the action enabling kernel and displays hotlinks by overlaying a transparent panel on top of the non-

9954_1

linkable media whereby the non-linkable media becomes interactive and hyperlinkable. The non-linkable media is not responsible to display hotlinks, track hotlinks or trigger actions on the hotlink The hotlink canvas performs the following functions: Query media current position and state; Forward media position and state to the action enabling kernel; Receive contending hotlink candidates from the action enabling kernel; Pick one hotlink among a set of hotlinks candidates; Compose hotlinks by receiving hotlink meta data from the action enabling kernel; Display range contours of hotlinks on the media object by overlaying a hotlink canvas; Perform an action specified; and Request the action enabling kernel to invoke an action. With the hotlink canvas, non-linkable media does not need to be aware of any hotlinks and corresponding action handling. The media can be plugged into the hotlink canvas framework and become interactive and hyperlinkable.

DESCRIPTION OF THE DRAWINGS

The invention will be further understood from the following detailed description of a preferred embodiment taken in conjunction with an appended drawing, in which:

Figure 1 is a representation of system incorporating HotMedia architecture and incorporating the principles of the present invention.

Figure 2 is a representation of a HotMedia file format for streaming rich media for use in the system of Figure 1.

Figure 3 is a representation of a client station for enriching non-linkable media representations by enabling an overlying and accompanying hotlink canvas of the non-linkable media for hyperlinking in the system of Figure 1.

Figure 4 is a flow diagram describing a method of operation for enriching otherwise non-linkable media for hyperlinking using an overlying, transparent, accompanying hotlink canvas in the system of Figure 1.

9954_1

Figures 5A & B are a representation of a hotlink canvas generated in a system of Figure 1 using the process of Figure 4.

DESCRIPTION OF PREFERRED EMBODIMENT

In Figure 1, a multimedia information system 100 implements HotMedia architecture and includes an HTTP or dedicated server 102 coupled to a multimedia network 108, typically the Internet. The server is further coupled to a storage medium 104 which stores presentation files in rich media created by standard authoring tools for delivery to a client station 106 coupled to the network and serving a user 110. The client station includes a presentation and user interaction logic unit 112 accessed by the user through a client terminal device 114. The presentation files 104 are supplied to the client station as streaming data on presentation bus 116. The user's interaction with the presentation data is returned to the server 102 as interaction data on bus 118. The presentation files and interaction data are exchanged in the network between the server and the client station using standard HTTP protocols. A user interaction tracking server 120 monitors the user's intentions, preferences and attitudes with respect to the presentation files and exchanges data with the server 102. The tracking server is more fully described in Serial No. _____ entitled "System and Method for Tracking User Interations and Navigation During Rich Media Presentations", filed _____ (SE9-99-011/1963-7344), supra.

Figure 2 shows a HotMedia presentation file 200 in a framework 201 which is essentially a sequence of frames types comprising header 202, thumbnail 204, meta 206, media 208, and an end of stream 210. The first frame is the header frame 202. The header frame is actually followed by the thumbnail frame 204. After these frames a sequence of other frames occurs and in no pre-mandated order or number. The header frame 202 is only one whose presence is mandatory in the format. The most degenerate yet useful version would have a header frame followed by a thumbnail frame 204. Media frames 208 appear in all other instances. Meta frames 206 are present only in cases where non-default behavior and configuration are desired on cases where hyperlinked action semantics are to be incorporated. All frames have a similar 12

9954_1

bytes initial section that enables a uniform procedure for their identification to a type and frame label as well as the determination of their size. The HotMedia file format is created to have minimum overhead and maximum modularity. The format makes it suitable for optimal delivery of a low bandwidth as well as for rich experiences over high bandwidth.

5

The header frame 202 includes a frame header 212 which provides information about frame size 214, types and flags 216 and frame labels 218. The header frame 202 also includes media information 220 containing definition on different media tracks that may be contained in the file 200 General information 224 in the header frame 202 provides the class name of the code that renders the media type on the client station. When HotMedia kernel code on the client side parses the media track information on the file header and if it encounters the specification of a class name, the media type field is ignored and the code specified by the class name is sent from the server in order to render the data of the track in question. The class name permits the introduction of whole new media types into the presentation. No changes or additions to the client kernel code are needed so long as the media objects implements the interfaces specified in the HotMedia framework 201 for first class members of a media object set.

The thumbnail frames 204 carry a minimalistic representation of the information shared by the HotMedia file framework 201. For example, if the information carried in the file were a description of merchandise, the thumbnail would carry a single image. Thumbnail frames make possible providing the user a quick introduction to the subject with minimal code and data transfer. Since users pay attention to detail only when the subject is of some interest, it would be wasteful for such data and codes to enable anything beyond the initial experience unless an interest is indicated. In addition to a frame header, the thumbnail frames also carry parameters for producing image pan and animation effects on the single image (all not shown).

Meta frames 206 carry specification of non-media specific information pertaining to enablement of hyperlinked actions from media context and tracking of interaction. Action semantics associated with media segments can be expressed in HotMedia files 200 using range 205 and action 207 subtypes. Range frames 205 are made up of range records that identify and label spatio-temporal segments of media. A spatial range can be described by a rectangular or

10

15

20

25

30

9954_1

elliptical contour. Temporal ranges can refer to a time range in milliseconds for a real time media-like audio or a count range for non-real- time media like animation. The range specification can also express a combination of space and time.

5   Action records 207 may be pre-programmed into the HotMedia file 200 for the execution of many types of action in the context of a presentation. Action frames can carry multiple action records all pertaining to a single actionable context, namely the presentation of media segment as specified by a range. Action records are the specification mechanism for all types of actions that can be triggered during the context of presenting a certain media segment while in a certain state

10   of the media object. Actions can belong to a variety of types that include LINK to specify hyperlinking; DISPLAY to specify the display of contours, cursor changes, etc; PLAY SOUND to create sound effects in various contexts; and TRACK to specify the tracking of interactions and events associated with a user experience. Also contained in an action record is the specification of an action trigger. The triggers for actions can be user interaction, presentation

15   state updates or player state updates. An example would be the coming in to view of a certain segment of a presentation, the movement of mouse into a spatio-temporal range of a presentation, the click of the left mouse button within the present temporal range of the presentation, the completion of loading of a segment of the media data and so forth. Although the range record specifying a media segment associated with most action triggers are of type

20   spatial, temporal or both, there are some situations where the range type is inconsequential. A range type DONTCARE is designed for such situations. An example is where the action is determined entirely by the internal state of the media object producing the completion of data loading trigger. The final piece of information contained in an action record is the data for action. The size of this data and its interpretation is different for each action type and subtype.

25   The data information is where for example, the target URL string is found when the action type is LINK and the text string, font and color information when the action type is DISPLAY.

       The media frame 208 includes a frame header 222 and in addition contains media bit stream data 209, initialization data 211, behavior data 213 and code 215. The behavior data or

30   initialization data define the behavior and configuration of a corresponding media player respectively. Media bit stream data belonging to a particular media track can be carried over

9954_1

multiple frames and these frames can be interleaved with meta frames and frames belonging to other media tracks. Media frames are identified by frame type and track identifier as carried in their header 222. The frame type and track identifier enable true multiplexed delivery and hence ideal for static or dynamic proportioning of various media tracks over limited bandwidth. Media

5   frames can also often contain codes 215 that render the media type on the client station. The code in turn can be physically present or could be a URL reference. When present, the code is separated from the file on the server side 102 (see Figure 1), and thereafter the code or data can be made available for on demand delivery. In the case of a URL reference to code, the code is fetched from a remote HTTP server to be similarly made available on demand.

10

       The end-of-stream frame 210 contains a frame header 226 and is a marker that signals the end of the entire presentation associated with a Hot Media file 200. Frames belonging to multiple media tracks and meta data can be multiplexed in the HotMedia file. The end of stream frame is valuable to the client station in recognizing the end of each HotMedia file data stream

15   which may otherwise be impossible when stream initialization data for each stream do not announce their respective lengths.

       Turning to Figure 3, the Client side station 106 will now be described in conjunction with processing the presentation file data stream 200. The server 102 provides a first streaming

20   HotMedia file 300 comprising a series of frames $301_1 \ldots 301_n$, previously described in conjunction with Figure 2. The frames $301_1 \ldots 301_n$ may contain meta frames with range and action subtypes. A HotMedia client master 303 receives the frames 300 and as soon as a meta frame is encountered with range and action subtypes, the master verifies to see if an action enabling kernel 307 has been created. If not, the master 303 obtains action enabling kernel code

25   304 and media object code 306 from the server 102 and instantiates the code 304 in an action enabling kernel 307 and creates an instance of the code 306 as a media object 308 on the client terminal 114 (see Figure 1). The client master 303 delivers the sequence of frames 300 to their proper owner. Media frames 309 are delivered to the media object 308 handling the media track. Meta frames 310 of the range 311 and action 312 types are delivered to the action enabling

30   kernel 307 and stored in range tables 313 and action tables 315.

The range table 313 and an action table 315 are updated based on meta frames included in the file 300. The action enabling kernel builds range and action tables out of the information contained in the range and action frames respectively. An entry is contained in the range table corresponding to each range record in the arriving frame. The information contained in the range

5   record is used as data for the entry. The table contains entries corresponding to ranges responding to any media track in the file 300 and no particular order is mandated. Similarly, an entry is created in the action table corresponding to each action record in the arriving action frame. The information contained in the action record is used as data for the entry. A cross-reference index is also added to the range table to enable easy access of the description of the

10 range with which the action in question is associated.

As the media object 308 receives media related data 306 from the client master and commences the rendering of the media it will also be receiving user input to mouse and keyboard interaction. The media object may also be undergoing internal state changes pertaining to

15 completion of data loading, commencement or rendering and so forth. The media object can signal the occurrences of this user input or state changes to the action enabling kernel 307. On receiving them the action enabling kernel regards the signals as triggers and as the occasion to verify the presence of actionable situation. When relevant to the media object in question, triggers can be associated with the display of every new image, mouse activity within an applet

20 space and media object state changes, such as the completion and loading of all images. It is entirely up to the media object which triggers it chooses to send to the action enabling kernel. Associated with the action enabling kernel are the action handlers 320, 321, 322 for responding to triggers having been initiated by the media object. The action enabling kernel may signal the master module 303 to fetch hotlinked files in response to some triggers such as a

25 REPLACEURL. The master module 303 provides data request 319 transmitted to the server once a request is submitted by the action enabling kernel. For example, action handler 320 may perform a VIEW change *on* the client terminal 114 (See Figure 1) in response to a change trigger. Action handler 321 may perform a PLAYSOUND. Action handler 322 may perform a LINK. Various other actions can be performed in response to triggers such as MOUSEMOVE,

30 REPLACEURL, etc. Triggers generate other triggers depending on the context. For example, a

10

MOUSE MOVE may cause a mouse to enter range or a mouse exit range depending on where the mouse was previously located.

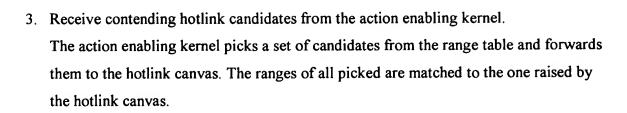Further details of the construction and operation of the Client station in the HotMedia
5   Architecture of Figure 1 are described in Serial No. _____ entitled "Personalizing Rich Media Presentations Based on User Response to the Presentation", filed _____ (SE9-98-028), supra.

In those instances where HotMedia files are non-linkable media representations, i. e. there are no meta frames containing range and action records for hotlink composition, the master
10   module 303 may enable the non-linkable media for hotlinking by fetching hotlink canvas code 331 from the server and instantiates the hotlink canvas object 332. In addition, a set of meta frames would be delivered to the action enabling kernel from the server for hotlink composition. The hotlink canvas creates a transparent panel 500 (See Figure 5) which is overlayed on top of the media object. By receiving hotlink composition data from the action enabling kernel, the
15   hotlink canvas displays hotlinks overlying the otherwise non-linkable media. The media is not responsible to display hotlinks, track hotlinks, or trigger actions on hotlinks. The media is only responsible to report its spatio-temporal position and current state whenever requested by the hotlink canvas. In effect, the hotlink canvas is a liaison between a media object and the action enabling kernel. The hotlink canvas performs the following functions:
20

1. Querying media current positions and state.
    In order to display hotlinks properly the hotlink canvas requests for the current status of the media object frequently. The current status of the media object comprises any user interaction by either mouse or keyboard input, current mouse position, i. e. (x, y)
25   coordinates, current temporal position in milliseconds, and the current state of the media object.

2. Forwarding media's position and state to the action enabling kernel.
    The hotlink canvas sends the current status of the media object to the action enabling
30   kernel to be verified if any hotlink(s) is triggered.

11

3. Receive contending hotlink candidates from the action enabling kernel.

   The action enabling kernel picks a set of candidates from the range table and forwards them to the hotlink canvas. The ranges of all picked are matched to the one raised by the hotlink canvas.

5

4. Selecting one hotlink among a set of hotlink candidates.

   The hotlink canvas may choose which hotlinks are triggered from candidates forwarded by the action enabling kernel.

10

5. Composing hotlinks by receiving hotlink meta data from the action enabling kernel.

   The hotlink canvas receives meta data from the action enabling kernel and stores them internally for hotlink composition and displaying.

6. Displaying range contours of hotlinks on the media object by overlaying a transparent panel.

15

   The hotlink canvas creates a transparent panel overlaying on top of the media object to display range contours of hotlinks without any awareness of the media object.

7. Performing an action specified.

20

   For example, if the action type is DISPLAY and sub-type is RANGE_CONTOUR, the hotlink canvas is ideally suited to perform this action.

8. Requesting the action enabling kernel to handle an action.

25

   Instead of performing an action, actions can be handled by one of action handlers provided by the action enabling kernel.

9. Unifying representations of hotlinks.

   The hotlink canvas transforms media information represented in the media space to a

30

   uniform representation. The hotlinks are identified by a uniform procedure

9954_1

13

independent of non-linear representations of media in the context of the hotlink canvas.

In Figure 4 a process 400 will be described in conjunction with Figures 1-3 for
5   constructing a hotlink canvas enabling a non-linkable media representation for hyperlinking in a
network. The process is entered at start point (S) after which a streaming rich media file 300 is
received from the server 102 in block 401. The streaming rich media file is received by the
HotMedia client master module 303 in block 403. The HotMedia client master module 303
starts to fetch the media object code 306, the action enabling kernel code 304 and the hotlink
10   canvas code 331 from the server 102 in block 405 while the rich media file is streaming to the
HotMedia client master. A test 407 is performed by the master module 303 to determine if any
media frames 309 have been received. A "yes" condition delivers the media frames to the media
object 308 in block 409, afterwhich the process returns to the entry point S. A "no" condition
transfers the process to block 411. A test is performed in block 411 to determine if the file
15   contains any meta frames. A "no" condition initiates a test in block 443 to determine if the
streaming rich media file from the server 102 is complete. A "yes" condition in block 443
transfers the process to entry point A which will be described hereinafter. A "no" condition in
block 443 transfers the process to block 401. Returning to block 411, a "yes" condition transfers
the process to block 413 in which the meta frames are delivered to the action enabling kernel
20   307. The action enabling kernel in a test 415 determines if any hotlinks are present in the meta
frames. A "no" condition transfers the process to entry point A. A "yes" condition transfers the
process to blocks 417 and 419. In block 417 hotlink composition data in terms of range and
action data from the action enabling kernel are entered in the hotlink canvas. In block 419 a test
is performed to display any spatial hotlinks in the meta frame. The test 419 transfers the process
25   to entry point A for a "no" condition. A "yes" condition transfers the process to block 421 in
which the hotlink canvas transforms the information reported by the media object into a uniform
representation. The representations of hotlinks in the context of the hotlink canvas are unified
and independent of non-linearities of the media. In block 423 range contours of the hot links are
displayed by overlaying a transparent panel as a hotlink canvas on top of the media object after
30   which the process transfers to entry point A.

13

At entry point A test 425 is performed to determine if any user interaction has occurred during the media presentation. A "no" condition cycles the process to entry point A. A "yes" condition transfers the process to block 427 in which the media object 308 reports its position and states with specific trigger types to the hotlink canvas. In block 429 the hotlink canvas

5   performs a non-linear transformation and interpolation to unify hotlink representations over non-linear media if necessary. In block 431 the hotlink canvas forwards the information to the action enabling kernel 307. In block 433 the action enabling kernel sends back a set of contending hotlink candidates to the hotlink canvas. The hotlink canvas picks one hotlink from a set of hotlink candidates based on the range and action data received from the action enabling kernel in

10   block 435. A test is performed in block 437 to determine if the corresponding actions are ideally suited to be performed by the hotlink canvas itself. A "yes" condition transfers the process to block 441 in which the actions are carried by the hotlink canvas after which the process transfers to entry point A. A "no" condition initiates a hotlink canvas request to the action enabling kernel to perform a specific action in block 439 after which the process transfers to entry point A. The

15   action enabling kernel determines if any actions need to be executed by specific action handlers 320-322. Those actions would be forwarded to action handlers 320-322 to be executed. The process remains in a waiting condition at entry point A after all actions have been performed until user interaction occurs in the media presentation after which blocks 425-441 are repeated or another streaming rich media file is received from the server after which the entire process is

20   repeated beginning with step 401 through 443 is performed.

Further details of unifying representations of hotlinks between non-linear media in block 421 and 429 are described in Serial No. _____ entitled "Method for Non-linear Transformation and Interpolation for Unifying Hotlinks Between Heterogeneous Media Representations", filed

25   _____ (SE9-99-004), supra.

In Figure 5A, the client terminal 114 (See Figure 1) displays non-linkable media representations (not shown) for incorporating hotlinks by an overlying transparent panel 501 as a hotlink canvas. The transparent panel is constructed by the hotlink canvas 332 and is juxtaposed

30   with respect to the non-linkable media representation. The hotlinks can be incorporated into the panel 501 in accordance with the process 400 described in Figure 4.

In Figure 5B, a transparent panel 503 can serve as a hotlink canvas to decouple hyperlinks embedded in media displayed on the terminal 114. The media with embedded hyperlinks need to be reconstructed in the form of a streaming rich media file 300 described in

5 Figure 3. The process includes parsing media frames, separating hyperlinks information, authoring media frames in accordance with the file 300 and transforming hyperlinks information to meta frames in accordance with file 300.

While the invention has been shown and described in connection with a preferred

10 embodiment various changes may be made therein without departing from the spirit and scope claimed and defined in the appended claims: